$_height, glyphlength =$
$+0.9ex, glyphshorten =$
$+-$
$0.1ex, drawingcode =$

I. INTRODUCTION

Most non-small-cell lung cancer (NSCLC) patients are inoperable due to locally advanced disease or distant metastases and thus radiation therapy (radiotherapy) becomes the main option for treatment of these patients. However, treatment outcomes remain relatively poor despite significant advances in the technologies of radiotherapy planning, image-guidance, and delivery [?]. It is conjectured that escalation of radiation dose is an option to improve treatment outcome results. For instance, a dose increment by 1 Gy can lead to 1% improvement in local progression free survival [?]-[?]. However, this has not always been demonstrated to be the case, as was learned from RTOG-0617 clinical trial results, where dose escalation has led to surprisingly negative results [?]. Though the specific causes of this negative finding are still being worked out, it is clear that dose escalation can not be employed using a one-size fits all approach to the patient population. While necessary for cancer treatment, radiotherapy provides cure but can also pose risks that need to be tailored according to each individual patients characteristics. For treatment of lung cancer, a major limiting constraint to dose escalation is the toxicity risk from thoracic irradiation that leads to radiation-induced pneumonitis (RP). RP causes cough, fever, etc, and it affects the quality of life for patients even if the local control (LC) of the tumor is assured. Therefore, an important question that the current studies are attempting to address is: can machine learning algorithms identify from patient characteristics an optimal dose schedule to render LC with maximally reduced RP in an individual patient? However, before attempting to address this challenging question, we need to demonstrate that machine learning algorithms can actually be taught to mimic clinicians decision making processes.

With the latest advances in machine reinforcement learning (RL) algorithms, which provide better dynamic learning options, we are poised to explore the feasibility of automated decision making for dose escalation in NSCLC patients. Traditional machine learning methods have witnessed increased applications in radiotherapy including quality assurance, computer-aided detection, image-guided radiotherapy, respiratory motion management, and now outcomes prediction [?]. However, traditional machine learning methods may lack the ability to handle the dynamics of highly complex decision making process in a clinical radiotherapy environment. For instance, our institutional protocol UMCC 2007-123 [?]-[?] defines dose escalation under a sophisticated adaptation policy (see Sec.??) towards improved treatment outcomes. Thus, it could be utilized as a suitable testbed to assess our proposed RL methods for automated radiation adaptation. The rationale for utilizing reinforcement learning in automating radiation dose adaptation is that it allows exploration of all possible paths into the future so that expected benefits and risks can be weighed into the decision making process. In an analogous fashion such as playing chess or board games, the decision maker needs to explore the consequences of the next moves and develop an optimal strategy to win the game, which in our case is controlling cancer while reducing treatment side effects. To realize this task within the complex radiotherapy environment, we developed dynamical procedures to utilize the existing historical treatment plans to represent the radiotherapy environment (Sec. **??**), where the states within this environment are defined as predictor factors of local control (LC) and radiation-induced pneumonitis (RP) responses.

In recent years, deep learning applications have gained success in variety of fields including video games, computer vision, and pattern recognition. A key factor in this success is that deep learning can abstract and extract high-level features directly from the data. This helps avoid complex feature engineering or delicate feature hand-crafting and selection for an individual task [?]. Recent studies have demonstrated that using a class of deep learning algorithms based on convolution neural networks can efficiently replace traditional feature selection in image segmentation while at the same time providing superior performance [?]-[?]. These strengths motivated Google DeepMind's incorporation of deep neural networks (DNN) into the known Q-learning search algorithm of RL [?], which enabled it to master a diverse range of Atari games with human-level performance using the raw pixels and scores as inputs [?]. The DQN algorithm has been shown to display actions similar to human instincts in playing these games. Such ability was demonstrated by AlphaGo when it dethroned the world champion of the ancient Chinese game *GO*, a 19×19 grid board game considered to have intractable ($316! \simeq 10^{678}$) possibilities. The sheer complexity of GO renders the ability to make human decisions from intuitive intelligence indispensable for playing properly and having a credible chance at winning. This study tends to pursue the characteristic of intuition-driven decisions in the DQN for mimicking and comparing clinicians dose adaptation decisions in treatment planning. However, there are millions of records with detailed moves of previously played games that could be used in training the DQN algorithm; this is a luxury that we do not possess in the clinical or the radiotherapy world. Therefore, we also incorporated new developments in deep learning for generating synthetic data to help meet the goal of training automated actions owing to the demand of high-sample-size requirement by the DQN. Specifically, we deployed 3 different DNNs to tackle several problems in building a machine learning approach for completing automation of clinical decision making for adaptive radiotherapy, see Fig. 1(b). The first DNN (GAN, Sec.II E) aims to generate sufficiently large patient data from existing small-sized observations for training the simulated radiotherapy environment. The second DNN is tasked to learn the radiotherapy environment, i.e., where and how states would transit under different actions (dose fraction modifications) based on the data synthesized from the GAN and real clinical data available, Sec. **??**. The third DNN is the innovative DQN itself responsible for prompt and accurate evaluation of the different possible strategies (dose escalation/de-escalation) and optimizing future rewards (radiotherapy outcomes). In contrast, classical RL methods such as the model-free temporal difference (TD) algorithm [**?**], which require a sufficiently larger number of observations to be sampled and high consistency in the states (variables) and actions (decisions), do not fit well with the complex, real clinical radiotherapy environment where the data are noisy and complete information may be missing as well as limited sample size. Moreover, clinical decisions are in general likely to be more subjective than objective. These are some of the hurdles that our approach based on the 3-component DNN design attempts to overcome. We believe that the proper integration of these three components based on deep learning is essential for building a robust RL environment for decision support in radiotherapy adaptation.

In previous work [**?**], Kim *et al.* developed a Markov decision process (MDP) from the perspective of analytical radiobiological response to compute optimal fractionation schemes in radiotherapy. The MDP design was based on delicate assumptions on the latent behavior of the tumor and the organs-at-risk (OAR) with respect to given dose. Several numerical simulations were presented and their behavior, based on the assumptions made, were discussed but no realistic clinical scenarios were evaluated. Another similar approach based on analyzing stochastic processes of reinforcement learning with TD techniques [**?**] was used to dynamically explore the transition probability with varying fractionation schedules. Based on simplified radiobiological assumptions, different reward (utility) functions were tested in preclinical cell culture data to nonuniformally optimize the prescribed dose per fraction [**?**].

Here, implementation details and network architectures are described and organized as follows. In Sec. II, we succinctly introduce the methods and rationale of our utilization; Sec. **??** demonstrates the results of the different components of our proposed approach and their benchmarking against real clinical protocol results. In Sec. **??** and Sec. **??**, we summarize our methods presentation as an integrated system and discuss future potential developments as well as the limitations of our current study.

II. MATERIALS AND METHODS

A. Overview

In our investigation to apply DQN for escalation of dose in NSCLC data, we first faced the obstacle of the absence of a well-characterized radiotherapy environment (i.e., the rules of the game) as shown in Fig. 1(a) and Fig. **??**. This is unlike the case of applying DQN to board games where complete information of the game rules are defined beforehand and also one can play the game repeatedly almost at no real cost. In the case of patient care in general or radiotherapy specifically, this would be ethically and practically prohibitive due to the consideration of patients' safety and the cost of time. To alleviate this difficulty, we developed a radiotherapy artificial environment (RAE), also referred to as the (approximate) transition DNN in Sec.**??** for simulating the radiotherapy treatment response environment. Due to the limited available sample size, we combined the GAN with the transition DNN to support the fidelity of reconstructing a RAE. As the GAN can generate synthetic patient data very similar in its characteristics to the real ones, we then trained the RAE with mixed data; the synthesized data by the GAN and the available real clinical data. After the reconstruction of the radiotherapy environment, we introduced the DQN agent (decision maker) into this environment to interact with it, as indicated in Fig. 1(b) and evaluated its performance by learning the adaptive behavior of a dose escalation clinical protocol conducted successfully in our institution and recently published in JAMA Oncology [**?**].

B. Datasets

We used historical treatment plans of 114 NSCLC patients for training our 3-component DRL for decision support of response-based dose adaptation. The patients had been treated on prospective protocols under IRB approval as described in [**?**]. All tumor and lung dose values were converted into their 2 Gy equivalents (EQD2) by an in-house developed software using the linear-quadric model with an α/β of 10 Gy and 4 Gy for the tumor and the lung, respectively. Generalized equivalent uniform doses (gEUDs) with various parameters a were calculated for gross tumor volumes (GTVs) and uninvolved lungs (lung volumes exclusive of GTVs). Blood samples were obtained at baseline and after approximately 1/3 and 2/3 of the scheduled radiation doses were completed. A total of 250 features including dosimetric variables, clinical factors, circulating microRNAs, single-nucleotide polymorphisms (SNPs), circulating cytokines, and positron emission tomography (PET) imaging radiomics features before and during radiotherapy were collected. Pre-treatment blood samples were analyzed for cytokine levels, micro RNAs (miRNAs), (a)How to let the DQN interact with limited-sample-sized data of the historical patient population treatment?

(b)A 3-component DNN solution is proposed to (1) generate synthetic data through GAN to (2) model the radiotherapy environment by the transition DNN that is used by (3) the DQN to make optimal decisions for adaptation of dose.

FIG. 1. A 3-component DNN solution to overcome limited sample size and model the radiotherapy environment for DRL decision making.

and single nucleotide polymorphisms (SNPs), which have been identified as candidates from the literature as related to lung cancer response. FDG-PET/CT images were acquired using clinical protocols and the pre-treatment and intra-treatment PET images were registered to the treatment planning CT using rigid registration. The image features analysis was performed using customized routines in MATLAB and the features included metabolic tumor volume, intensity statistics, and texture-derived metrics [? ?]. Part of this population, with dose adaptations at ∼ 2/3 of the way through treatment as served by Protocol UMCC 2007-123 [?]-[?], are described in Sec. **??**. Nine predictive features, defined in (**??**) with characteristics described in TABLE. **??**, were selected for modeling the RAE. These features are related to LC and RP2 responses based on Markov blankets and Bayesian analyses as detailed in [?] and briefly reviewed below.

C. Variable Selection for simulating radiotherapy environment

In order to define the radiotherapy environment via a large-scale variable list, we used techniques based on Bayesian network graph theory, which allows for identifying the hierarchical relationships among the variables and outcomes of interest. The approach we used is based on identifying separate extended Markov blankets (MBs) for LC and RP2 from the above high-dimensional dataset of 297 candidate variables. An MB of LC (or RP2) is the smallest set containing all variables carrying information about LC (or RP2) that cannot be obtained from any other variable (inner family); then for each member in the blanket of LC (or RP2), a next-of-kin MB for this member was also derived using a structure learning optimization algorithm [?]. The algorithm combines efficient graph-search techniques with statistical resampling for robust variable selection [?]. The selected variables by this approach are summarized in (**??**). It should be emphasized that the purpose of this step is to provide an approximate radiotherapy environment that would allow simulating transitions between its states when the DQN agent is making decisions.

D. Deep Neural Networks

We mainly utilized deep neural networks (DNNs) for our proposed DRL approach and the main notations used are summarized here for convenience. Denoting data $\{\mathbf{x}_i\}$ with labels $\{\mathbf{y}_i\}$ such that $\mathcal{X} = \{(\mathbf{x}_i, \mathbf{y}_i)\,|\mathbf{x}_i \in \mathbb{R}^n,\ \mathbf{y}_i \in \mathbb{R}^m,\ i = 1, \ldots, N\}$, a DNN finds a function $f_{\text{DNN}} : \mathbb{R}^n \to \mathbb{R}^m$ to weave through the data such that $f_{\text{DNN}}(\mathbf{x}_i) \cong \mathbf{y}_i$ as much as possible via the utility of three distinct components: *neurons* $z_i \in \mathbb{R}$, *layers* of k neurons $\mathbf{z} = (z_1, \ldots, z_k)$, and *activation functions* σ, see Fig. **??** (left). If a DNN has layers $j = 0, \ldots, \ell$, each of which has n_j neurons, then $j = 0$ and ℓ would denote the first (input) and final (output) layer, respectively. An activation function $\sigma : \mathbb{R}^{n_{j-1}} \to \mathbb{R}^{n_j}$ connecting the neurons of the $(j-1)^{th}$ layer $\mathbf{z}^{(j-1)} \in \mathbb{R}^{n_{j-1}}$ and those of the j^{th} layer $\mathbf{z}^{j} \in \mathbb{R}^{n_j}$ would satisfy:

$$\mathbf{z}^{(j)} = \sigma\left(\Theta^{(j-1)} \cdot \mathbf{z}^{(j-1)} + \mathbf{b}^{(j-1)}\right) \tag{1}$$

where $\Theta^{(j-1)} \in \mathbb{R}^{n_j \times n_{j-1}}$ and $\mathbf{b}^{(j-1)} \in \mathbb{R}^{n_{j-1}}$ represent the unknown weights and biases to be estimated. A typical choice of σ is a *sigmoid* or a *rectified linear unit* (ReLU), where we empirically choose $\sigma = eLU$ [**?**] in this study for better convergence. Our best parameters $\{\Theta^{(j)}, \mathbf{b}^{(j)}\}_{j=0}^{\ell-1}$ are then derived from the forward dynamics and backward (error) propagation resulting from the DNN loss function [**?**]:

$$\mathcal{L}(\Theta, \mathbf{z}, \lambda) = \frac{1}{2}\|\mathbf{y} - \mathbf{z}^{(\ell)}\|^2 - \sum_{j=1}^{\ell}\left\langle \lambda^{(j-1)}, \mathbf{z}^{(j)} - \sigma\left(\Theta^{(j-1)} \cdot \mathbf{z}^{(j-1)} + \mathbf{b}^{(j-1)}\right)\right\rangle \tag{2}$$

where $\lambda^{(j-1)} \in \mathbb{R}^{n_{j-1}}$ are the Lagrange multipliers at layer $j-1$ to preserve layerwise information (1).

In this study, we primarily rely on the universality of DNNs to model the dynamic complexity hidden in the radiotherapy data. This universality refers to the capability of a neural network to approximate any continuous function (on a compact subset of $\mathbb{R}^n$) with suitable activation functions [**?**]. Due to limited patient sample size, we implemented random dropouts on neurons to efficiently mitigate overfitting [**?**] throughout. In such scenario, randomly selected neurons are assigned zero weights, which is a form of regularization to prevent the network from over-adaptation (overfitting) to the data during training process.

E. Generative Adversarial Nets

To alleviate the problem of small sample size in clinical datasets when modeling the complex state transitions in a radiotherapy environment, we utilize generative adversarial nets (GANs) [**?**] to synthesize more radiotherapy *patient-like* data. A GAN consists of two neural nets, one of which is generative (G) and responsible for generating synthetic data, and the other one is discriminative (D), which tries to measure the (dis)-similarity between the synthesized and real data as shown in Fig. 2. The basic underlying idea is simple: by learning to confuse D, G can get more sophisticated in generating similar data through the following setup.

FIG. 2. GAN is used to generate new data, where G asks D to verify the authenticity of the data source. From latent points $\mathbf{z}$, generated patients are synthesized as $\widetilde{\mathbf{x}}$ in G. With $\mathbf{y} = (\mathbf{x}, \widetilde{\mathbf{x}})$ mixing with real and the generated patient data, D is trying to verify its source.

Denote the space $\mathcal{X} \supseteq \{\mathbf{x} \in \mathbb{R}^n\}$ containing the (original) dataset with distribution $\mathbf{x} \sim P_{\text{data}}$, and there is a latent space $\mathcal{Z} = \{\mathbf{z} \in \mathbb{R}^m\}$ with a prior distribution $\mathbf{z} \sim P_{\text{prior}}$, where in our case a Gaussian distribution is assumed. The generative network $G : \mathcal{Z} \to \mathcal{X}$ tries to learn a map from $\mathcal{Z}$ to $\mathcal{X}$ such that an induced probability distribution $P_G = P_{\text{prior}}(G^{-1}) \cdot |\det\left(\frac{\partial G^{-1}}{\partial \mathbf{x}}\right)|$ on $\mathcal{X}$ is close to the original P_{data}. The discriminative network $D : \mathcal{X} \to \mathbb{R}$ then simultaneously learns to discriminate observations from the true data and the synthesized data generated by G. In general, G aims at generating indistinguishable data to confuse D, whereas D attempts to distinguish the data produced by G or not. They interact with each other in a competitive sense, hence the name GAN. The adversarial characters of D and G are created via the loss function of two-player mini-max game:

$$\min_G \max_D \mathcal{L}(D, G) = \min_G \max_D \mathbb{E}_{\mathbf{x} \sim P_{\text{data}}(\mathbf{x})}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z} \sim P_{\text{prior}}(\mathbf{z})}\left[\log(1 - D(G(\mathbf{z}))\right], \tag{3}$$

where $D(\mathbf{y}) = \begin{cases} 1, & \mathbf{y} \text{ is real} \\ 1/2 & \mathbf{y} \text{ is indistinguishable} \\ 0, & \mathbf{y} \text{ is generated} \end{cases}.$

Subsequently, we introduce the algorithm for generating synthetic data for building our DRL for dose adaption.

F. Deep Q-Networks

We are applying reinforcement learning to mimic how physicians decide dynamically on the dose fraction trade-offs needed to prescribe to a certain patient. In reinforcement learning, there is the environment (an MDP) and an agent (an optimal action search algorithm). An agent takes charge of delivering actions $a \in \mathcal{A}$ in an environment, which is a world described by the various states $s \in S$ in the environment. Upon a decision made under a current state $\pi : S \to \mathcal{A}$, an agent receives corresponding reward R and gets promoted to another state. The transition between states and rewards R are feedback for the agent to perceive how to optimize its subsequent strategy for future actions. In our setting, an artificial agent would provide a second opinion or take place of a physician to deliver actions. Specifically, in this study, we will evaluate the required dose per fraction (adaptation) in the second period of a dose-escalation radiotherapy treatment course. This agent will then interact with the radiotherapy artificial environment (RAE) reconstructed by the transitional DNN, Fig. 1(b), and adjust its own adaptation strategy based on received feedback.

Reinforcement learning is essentially formulated as a Markov decision process (MDP), denoted by $(S, \mathcal{A}, P, \gamma, R)$, where $S = \{(x_1, \dots, x_n) \in \mathbb{R}^n\}$ be the space of states, $\mathcal{A}$ is the collection of actions, $R : S \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $P : (S \times \mathcal{A} \times S,\, \Omega) \to [0, 1]$ is the transition probability function between two states under an action $a \in \mathcal{A}$ with Ω a σ-algebra of $S \times \mathcal{A} \times S$ that naturally induces a conditional probability $P_{sa}(t) \equiv \mathrm{Prob}(t\,|s, a) \equiv P(s, a,\, t\,)/P(s, a)$ on space of next states $t \in S$ from previous observation $(s, a) \in S \times \mathcal{A}$. A sequence of actions acting on an initial state $s_0 \in S$ leads to the dynamics of an MDP:

$$s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} \cdots .$$

The Q-learning search algorithm is a common method to find an optimal policy given an MDP or an RAE in our case, where a Q-function is defined as the average discounted sum of rewards R in all future steps from current state s

under a policy $\pi : S \to \mathcal{A}$ as in (4). The expectation value is considered in the sense of computing all possible paths starting from current state s to represent all possible benefits received in the future. A discounting factor $0 \le \gamma \le 1$ diminishes how we perceive future profits, providing a trade-off between the importance of immediate reward versus future ones, i.e., short-term responses versus long-term outcomes.

In Q-learning, an optimal policy $\pi^* : S \to \mathcal{A}$ is defined such that $Q^{\pi^*} = \max_\pi Q^\pi$ is satisfied when the value iteration scheme is adapted for computation. Via the Bellman's equation of *off-policy*, the estimation of optimal Q^{π^*} is converted into an iterative sequence $\{\widetilde{Q}_i\}_{i=1}^{\infty} \to Q^{\pi^*}$ defined by [?]:

$$Q^\pi(s, a) = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k\, R(s_k, \pi(s_k))\Big|\pi,\, s_0 = s,\, a_0 = a\right] \tag{4}$$

$$\widetilde{Q}_{i+1}(s, a) = \mathbb{E}_{t\sim P_{sa}}\left[R(s, a) + \gamma \max_{b\in A} \widetilde{Q}_i(t, b)\right]. \tag{5}$$

Upon the contraction mapping theorem [?], the convergence is reached at the unique fixed point as $i \to \infty$,

$$\widetilde{Q}^*(s, a) = \mathbb{E}_{t\sim P_{sa}}\left[R(s, a) + \gamma \max_{b\in A} \widetilde{Q}^*(t, b)\right]. \tag{6}$$

It can be noticed that computation of (5) can quickly become cumbersome when the cardinality $|S|$ or $|\mathcal{A}|$ is large. A recent solution proposed by Google DeepMind in [?] and [?] was to evaluate the Q-function efficiently using supervised learning by DNNs by $\widetilde{Q}_i = Q^{\Theta_i}_{\mathrm{DNN}}$, where Θ_i denotes the weights of DNNs in (1) at i^{th} iteration with a sequence of loss functions $\mathcal{L}_i(\Theta_i)$ to be minimized where:

$$\mathcal{L}_i(\Theta_i) = \mathbb{E}_{(s,a)\sim\rho}\left[\left(\mathbb{E}_{t\sim P_{sa}}\left[R(t, a) + \gamma \max_{b\in A} Q^{\Theta_{i-1}}_{\mathrm{DNN}}(t, b)\right] - Q^{\Theta_i}_{\mathrm{DNN}}(s, a)\right)^2\right]. \tag{7}$$

where ρ is the probability distribution over policy sequences s and actions a also called the *behaviour distribution*. The loss function (7) can be understood to pursue a DNN sequence $\{Q^{\Theta_i}_{\mathrm{DNN}}\}_{i=1}^{\infty}$ such that $\{Q^{\Theta_i}_{\mathrm{DNN}}\}_{i=1}^{\infty} \to \{Y_i\}_{i=1}^{\infty}$ since (7) indicates:

$$\mathcal{L}_i(\Theta_i) = \mathbb{E}_{(s,a)\sim\rho}\left[\left(Y_i(s, a) - Q^{\Theta_i}_{\mathrm{DNN}}(s, a)\right)^2\right]. \tag{8}$$